8
COMPUTER-ASSISTED SEARCH

The preceding four chapters addressed successive stages of the search and detection process: Chapter 4, evaluation of naval sensors, given target detectability and position and environmental factors; Chapter 5, cumulative evaluation as sensors are applied over time; Chapter 6, the tactically useful MOE sweep width; and Chapter 7, evaluation of some types of search plans. This chapter discusses methods usable on a desktop computer to present to a search planner a probability map of target position at a user-chosen time. These implementations of search methodology are an important category of tactical decision aids known as **computer-assisted search (CAS)**. CAS programs have been applied with much success in exercises and operations in ASW and in Coast Guard search and rescue.

CAS usually employs methods from all of the preceding four chapters. The principal new methods in this chapter are (a) modeling target motion as a simply-described probabilistic, i.e., **"stochastic" process** (defined below), (b) application of Bayes' theorem from probability theory to account for unsuccessful search, i.e., **"negative information"** (defined below), and optimal allocation of search effort. Both (a) and (b) are important to preparation of a current or future probability map of target position. While the emphasis is on moving targets, the discussion begins with the much easier stationary-target case.

Usually CAS systems are also updated for "positive" information, i.e., information provided by target contacts with uncertain position and credibility, and present a recommended search plan in addition to the descriptive probability maps. Positive information updating is beyond the scope of the present treatment.

The target motion model employed in 803 is sometimes called a **track bundle** approach. This is a Monte Carlo method and has been a much-used approach to motion modeling in CAS systems. Analytic motion models have also been used in CAS, notably Markov chains. Markov chains as such are easily described, but they typically use a very large number of states to model target motion realistically. This poses a computation challenge that can be overcome in a significant class of cases by recursion under Bayesian filtering (not treated here).

To convey some of the basic concepts, section 801 gives an elementary example of search for a *stationary* target. Construction of a prior distribution of target position, Bayesian updating for negative information, and optimal allocation of search effort, based on the Bayesian updating method, are illustrated.

The main elements of a CAS system for a *moving* target are outlined in 802. These are a **prior probability map** of target position; a model of target motion; updates of the probability map for target motion, for negative information (which requires a model of cumulative detection probability), and for positive information; and search plan recommendations.

The usual Monte Carlo approach to CAS is illustrated in 803 by an idealized elementary example in which probabilistic target motion is represented by a bundle of only 16 tracks. The probability of occurrence of each track is derived from simple assumed distributions. The probabilistic behavior is quite visible. Motion updating is done by moving the target along each track according to the effect of that track, without changing track probabilities. Updating for negative information is done by changing track probabilities rather than geographic cell probabilities as in the stationary target case. These are important features of this Monte Carlo approach to CAS.

An algorithm for optimal allocation of search effort in space and time against a moving target is given in 804. It is illustrated by an example, which also shows that it need not be optimal to allocate myopically at each instant, without considering later instants. However, myopic search is usually fairly close to optimal.

801 Stationary Target

This section treats an elementary example of search for a stationary target, to illustrate some basic CAS concepts that are used in planning a search for either moving or stationary targets. The topics illustrated are map discretization, multi-scenario construction of a prior distribution (prior probability map) of target position, Bayesian updating for negative information, and optimal allocation of search effort.

Map discretization. In CAS applications, geographic positions in a search region are always shown by dividing the region into a rectangular array of discrete cells. A simple example of a 3×3 array of such cells is shown in Figure 8.1. Here the cells are indexed 1, 2, 3 in latitude and the same in longitude. They could just as well be indexed by mid-latitudes and mid-longitudes of the cells. A CAS program usually chooses cell size, but it is desirable and usual to let the user change this choice. The main factors influencing this choice are accuracies in placement of search effort and in estimation of positional probabilities. It is usually desirable to **smooth** the displayed probabilities in a map that has a realistic number of cells. This might be done by averaging each interior cell with its (presmoothing) neighbors, using suitable weights.

Multi-scenario construction of a prior. In Figure 8.1 (a) and (b), two scenarios, I and II, are assumed. Each **scenario** is a postulation as to what caused the target to be wherever it is. Associated with each scenario is a distribution of target position that has been derived from that scenario. Preferably this derivation is based on the information of the scenario as to *causes* of the target position; that is called a **causal** derivation. The distribution is given by assigning a number between 0 and 1 to each cell, with these numbers adding to 1. Each assigned number is the probability, before the search begins, that the target is in that cell, providing that scenario is valid. Also associated with each scenario is a number between 0 and 1 called the **scenario weight**. The scenario weights (here two) also add to 1. Each weight is an estimate of the probability that that scenario is valid. It is usually arrived at by consulting opinions of experts and may be regarded as a "subjective probability."

FIGURE 8.1. MULTI-SCENARIO CONSTRUCTION OF PRIOR DISTRIBUTION OF TARGET POSITION (STATIONARY TARGET)

(a) Scenario I
Weight = .7

		Longitude Index		
		1	2	3
	1	.1	.3	.0
Latitude Index	2	.3	.3	.0
	3	.0	.0	.0

(b) Scenario II
Weight = .3

		Longitude Index		
		1	2	3
	1	.00	.00	.00
Latitude Index	2	.00	.25	.25
	3	.00	.25	.25

(c) Composite Scenario

		Longitude Index		
		1	2	3
	1	.070	.210	.000
Latitude Index	2	.210	.285	.075
	3	.000	.075	.075

The composite distribution in Figure 8.1(c) is obtained by combining the single-scenario cell probabilities according to the scenario weights. E.g., the composite for latitude index 2 and longitude index 2 is

$$.3 \times .7 + .25 \times .3 = .285.$$

The distribution is also called the probability map of target position or probability map for short. In particular, it is the prior probability map, further abbreviated as the prior.

FIGURE 8.2. SCORPION SEARCH PRIOR DISTRIBUTION OF TARGET POSITION

NOTE: Shading indicates magnitude as follows: ***x***	$0 \le x \le 10$	***x***	$10 < x \le 100$
* Indicates location of Scorpion. ***x***	$100 < x \le 1000$	***x***	$1000 < x \le 10,000$

Convert numbers to probabilities by dividing by 10,000.

																	5	7	1	
				A	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*I*	*J*	*K*	3 *L*	11 *M*	14 *N*	24 *O*	6	
			1											5	26	35	22	26	9	1
			2										18	46	74	42	18	10	4	2
			3								8	60	140	99	45	20	4	2	1	1
			4	2	21	137	16	7	1	20	215	239	105	30	5	3	1	1	1	
			18 **5**	40	46	747	30	12-50	205	571	277	38	5	2	1	1	1			
		14	326 **6**	3	1	28	31	63	*85	62	1	8	7	10	7	3	4			
	1	359	175 **7**	174	10-96	282	245	82	71	65	35	27	9	12	6	5	4			
		24	25 **8**	42	82	297	230	129	115	61	33	14	14	10	6	2	5	1		
		17	25 **9**	20	20	20	19	55	99	46	30	14	15	3	5	1	6			
		2	13 **10**	14	25	20	24	45	34	27	19	15	5	7	5	5	1			
			11	7	13	12	9	1	3	3	11	14	5	4	3	2	1			
			12							1	4	4	10	5	4	1				
												1	3	2						
													3	2						

Figure 8.2 presents a real-life prior, from the 1968 *Scorpion* search. It was constructed as a weighted composite of nine single-scenario priors, as above, and of course is much more complicated than Figure 8.1. Among the scenarios were (I) *Scorpion* struck a sea mount and glided to the bottom, and (II) a torpedo turned active in a tube and *Scorpion* was unsuccessful in her maneuver prescribed for that emergency. For each of these and various other scenarios, a position distribution on the ocean bottom was causally derived, and scenario weights were obtained by expert opinion. Figure 8.2 ensued. The remains were found within a submarine length of the highest-probability cell,

after a five-month search. For planning purposes several probability distributions of time-to-detection, measured in search time on the bottom, were derived from this prior using different assumptions regarding such factors as the quality of navigation. The mean for each distribution was computed. The actual search time of 43 days turned out to be within the interval defined by the computed means, which was 35 to 45 days.

FIGURE 8.3. APPLICATION OF SEARCH EFFORT (STATIONARY TARGET)

		Longitude Index		
		1	2	3
	1	.0	.3	.0
Latitude Index	2	.0	.4	.0
	3	.0	.0	.0

Negative information update. An update for negative information will now be illustrated. Suppose search effort is applied to the 3×3 array of cells resulting in the probabilities of detection shown in Figure 8.3. The detection probabilities given in Figure 8.3 are conditional probabilites, conditioned on the presents of the target in the cell being searched, and are indicative only of quality and amount of search effort, and tell nothing about target location. The latter remains as in Figure 8.1(c).

Suppose this effort is unsuccessful. What is the new, i.e., **posterior**, probability map? It is known that the target is now less likely to be in the cells searched than it was and consequently it is more likely to be in the other cells. That is valuable information and should not be ignored, but how does one adjust the prior probability map accordingly? The answer is to apply Bayes' theorem (see Appendix A). This may be done in spreadsheet fashion as follows (of course, a CAS program would do this for a user):

[1] Cell lat/lng index (*i*, *j*)	[2] Pre-search (prior) probability target is in (*i*, *j*)	[3] Search *failure* probability *if* target is in (*i*, *j*)	[4] [2]×[3]	[5]=[4]/*S* Posterior probability target is in (*i*, *j*)
(1,1)	.070	1.0	.070	.085
(1,2)	.210	.7	.147	.179
(1,3)	.000	1.0	.000	.000
(2,1)	.210	1.0	.210	.255
(2,2)	.285	.6	.171	.208
(2,3)	.075	1.0	.075	.091
(3,1)	.000	1.0	.000	.000
(3,2)	.075	1.0	.075	.091
(3,3)	.075	1.0	.075	.091
	1.000		*S* = .823	1.000

Column [2] is the prior. Column [3] is usually called the **likelihood** of the observed event given the inferred event. Column [4] is proportional to the **posterior** distribution, which reflects real-world observations. Until this information is output to a user, it may be left in this unnormalized form. That is the reason the term "weights" is used. If probabilities are needed column [4] is normalized. This is done by dividing column [4] by its sum, *S*, resulting in the posterior, column [5]. The posterior is shown geographically in Figure 8.4.

FIGURE 8.4. PROBABILITY MAP UPDATED FOR NEGATIVE INFORMATION

Probabilities that a given cell contains the target given that the effort in Figure 8.3 did not succeed in detection.

		Longitude Index		
		1	2	3
	1	.085	.179	.000
Latitude Index	2	.255	.208	.091
	3	.000	.091	.091

The foregoing implements Bayes' theorem for this application. For cell (i, j) this result is given by the formula

$$\text{posterior } \Pr\{\text{target in } (i, j) \mid \text{no detection}\} = \frac{\Pr\{\text{no detection} \mid \text{target in } (i, j)\} \times \text{prior } \Pr\{\text{target in } (i, j)\}}{\text{normalizing factor}}.$$

Here the first probability on the right side is the likelihood factor. The normalizing factor (*S* in the above table) is the probability that the search fails.

Optimal search against stationary target – example. The probabilities in Figure 8.3, and hence those in Figure 8.4, depend on the amount of search effort applied to the various cells. Usually a search planner can choose among various allocations of effort, cell by cell, and would prefer to do so optimally.

To illustrate this, suppose the search is by an aircraft looking for a life raft assumed to be stationary. Suppose the nature of the search is such that the cumulative detection probability through search time *t*, $F_d(t)$, is given by the random search formula (7-2):

$$F_d(t) = 1 - e^{-wvt/A},$$

where *w* is sweep width, *v* is search speed, and *A* is the area of the cell searched. Assume $w = 30$ nm, v = 200 knots, and $A = 20{,}000$ sq nm. Then

$$F_d(t) = 1 - e^{-.3t}.$$

(It might be that *w*, and accordingly the coefficient .3 in $F_d(t)$, change from cell to cell, but assume here that they do not.)

Referring to Figure 8.1(c), it is clear that initial effort should be applied to cell (2, 2), since it has the highest probability of containing the target, .285. The question is how long should the search remain in (2, 2) before putting effort into (1, 2) and (2, 1), which have the second highest prior probability of containing the target, .21? One might apply the Bayesian algorithm to find the value of *t* which drops the posterior probability in (2, 2) to .21. However, that ignores the fact that as the posterior probability falls in (2, 2), it rises in (1, 2) and (2, 1). The solution, of course, is to find the t where these falling and rising posterior probabilities meet. Noting that cumulative *failure* probability is exp(-.3t) and setting the posterior probabilities of cells (2,2) and (1,2) equal to each other gives

$$\frac{.285e^{-.3t}}{S} = \frac{.21}{S}$$

and solving for t gives

$$t = \frac{\ln(.21/.285)}{-.3} = 1.02 \text{ hrs.}$$

Thus after 1.02 hours the effort should be divided equally among (2, 2), (1, 2), and (2, 1), since all three have the same probability (which has not been calculated) at that point.

This procedure can be continued until all cells which initially had non-zero probability of containing the target have equal probability, and accordingly, subsequent search is divided equally among them.

Optimal search against stationary target – general algorithm. The foregoing procedure may be restated as an algorithm in more general form as follows.

Let C be a finite set of cells, one of which contains the target. An amount of search effort is available, which the searcher may divide among the cells in C. Assume exponential effectiveness of search effort in the sense that there is a $\beta > 0$ such that if z is an amount of search effort applied to the cell containing the target, then 1 - exp($-\beta z$) is the probability that detection results from z. (The dimensions of β and z must be such that βz is dimensionless.) The initially available effort is fully applied in a sequence of application steps.

An application step begins with the total then-remaining effort z and for c in C, the probability $p(c)$ at that point that c contains the target; it may just as well be assumed that $p(c) > 0$, since if $p(c) = 0$, no search effort should be applied to c. If when the step begins, all cells have the same p value, then divide z equally among them. Otherwise choose c_1 and c_2 in C such that $p(c_1)$ and $p(c_2)$ are respectively the highest and second-highest (different) values of p. Then to every c in C whose p value is $p(c_1)$ apply

$$-\frac{1}{\beta}\ln\left(\frac{p(c_2)}{p(c_1)}\right)$$

amount of search effort, unless this would exhaust z, and none to other cells. If exhaustion would occur, then instead divide z equally among the cells with probability $p(c_1)$ and the allocation is complete. Whether or not exhaustion occurs, compute

$$S = \sum_{c \text{ in } C} p(c)e^{-\beta x(c)},$$

where for each c, $x(c)$ is the effort applied to c in this application step. Then S is the non-detection probability for this application step. If the target has not been found and effort remains, then p(c)exp($-\beta x(c)$)/S is c's containment probability at the start of the next application step (S is the normalizing factor). Repeat the procedure until the effort is exhausted. This results in an optimal allocation. The failure probability for the entire procedure is the product of the S factors over all the application steps. If the total-procedure failure probability is not needed, there is no need for normalization – numbers proportional to containment probabilities suffice to produce the allocation.

Note that this algorithm is "myopic," i.e., one always searches in the cell(s) of highest current probability. This optimizes the *currently* available effort without regard to what additional effort may become available thereafter. E.g., suppose a planner were initially allowed 4 units of search effort and planned accordingly by the above method. Then suppose the planner is allowed an additional 3 units of effort. Might it then be wished that the first 4 units had been used differently in light of having a total of 7 units available? The answer is no – myopic search is optimal. This statement depends very much on the target being stationary. As will be seen in 804, if the target is moving, myopic search need not be optimal.

802 Principal Requirements for Moving Target CAS

As a lead-in to CAS for *moving* targets, the requirements for such a system and means by which these requirements can be met are noted succinctly.

Prior map. A CAS analysis begins with a prior map (probability map of the target's position). This may be constructed as a weighted sum of single-scenario maps, each derived causally, preferably. Typically it begins with a single report of a target's approximate location at a particular time. Alternatively, it may be derived from historical analysis of past target habits.

Target motion model. Target motion must be described in probabilistic terms. This inevitably means that it is given as a stochastic process. Motion models are illustrated here in a Monte Carlo framework.

Most CAS systems have used Monte Carlo target motion models consisting of a bundle of (typically 500) target tracks, each labeled with the probability that it is (approximately) the actual track. A method is needed for the CAS user to construct this bundle and the associated probabilities from a menu of building blocks and the user's knowledge or assumptions of target behavior. Alternatively, the bundle of tracks may be constructed from historical analysis, and this may be done simultaneously with construction of the prior map. Whether a structure of building blocks with prior assumptions or historical analysis is used, it is usually most efficient to construct the track bundle by random sampling, as described below, after Figure 8.6.

The track probabilities in a Monte Carlo model are converted at any time to geographic cell probabilities by adding for each cell the probabilities of the tracks with positions in that cell.

Updated maps. The main object of CAS is to produce a probability map of target position *at a user-chosen time* and to do so from time to time. To do this, updates are needed for target motion and negative information. CAS systems also update for positive information, i.e., contact reports of uncertain position and credibility, which is not addressed here. When one utilizes positive and negative information jointly to estimate target state, notably position and velocity, one is engaged in **tracking**.

Target motion is updated in track bundle modeling rather simply: The tracks remain fixed, and in motion updating without new information the track probabilities remain fixed. For each track, the target position is simply moved along the track to the position of the chosen time. In the illustration of 803, the track mechanism is deterministic, but in some systems it is probabilistic. Track speeds as well as courses may differ from track to track. An update for motion under an analytic model follows the mechanism of the model.

Updating for negative information is done by application of Bayes' theorem. In Monte Carlo modeling this is best done by updating the *track* probabilities, and going from there to geographic cell probabilities.

Negative information updating also requires an estimate of the effectiveness of the (unsuccessful) search effort applied. This in turn requires a model of cumulative detection probability (cdp), a subject discussed in Chapter 5.

Optimal search plans. For a CAS system to compute optimal search plans may be considered highly desirable rather than a necessity. If the user is provided with good probability maps, guidance to search planning is at hand – search in the cells of highest detection probability (myopic search). However, it may not be practical to place the next increment of search effort on just the high probability locations, so an optimal practical plan is desired also. It is also often possible to improve significantly on myopic approaches. Most CAS systems provide methods of doing both of these things. The methods include selectively exhaustive examination of a reasonable set of alternatives, optimal placement of a rectangular application of search effort, and more sophisticated algorithms to compute optimal allocation of effort in time as well as space. For the latter, see 804. The theory of optimal allocation of effort is much better developed than the theory of optimal choice of path by which to deliver that effort.

803 Simplified Illustration of Moving Target Monte Carlo CAS

This section illustrates the principal method used in Monte Carlo CAS against moving targets. It does so by a simplified example of target motion, application of which captures the main principles involved. This approach is an excellent example of tactical decision aid modeling in that the modeling ideas and computer implementation are intimately and effectively intertwined. This point applies in particular to updating track weights rather than cell position probabilities, as described below.

Construction of target motion model. Figure 8.5 gives assumptions from which one can quickly build a model of target motion in a simplified search example.

Suppose there are two scenarios, I and II, representing two principal courses of action by the target. These have respective probabilities of occurrence .6 and .4. For each scenario, assumptions are made of target initial position, course, and speed. For each of these there are four possibilities, but for a given scenario only two positions, two courses, and two speeds have non-zero probability. It is assumed here that course and speed remain fixed once chosen. Realistic implementations provide for course changes and much richer distributions of initial course, speed, and position and of scenarios than the two-point distributions assumed here.

Each choice of scenario, initial position, course, and speed, all four being deemed independent, determines a sample target track. There are 16 such tracks and they are tabulated in Figure 8.5 along with probability of occurrence in the last column. For example, the prior probability that track 5 occurs is

$$.6 \times .3 \times .8 \times .4 = .058,$$

as seen from the four two-point distributions. A geographic plot of these 16 tracks is given in Figure 8.6, which identifies the four possible initial positions *A*, *B*, *C*, and *D*. For each start point and course, there is a track for each of two speeds, and these are plotted close to each other as dashed and solid lines. Each track is labeled with probability of occurrence. The 16 tracks together with their probability labels constitute a "stochastic process." This is one type of definition of the stochastic process concept: a probability distribution over a set of "sample paths." In operational CAS systems the bundle would contain 500 or more tracks, each generally having more complexity than the 16 illustrated here. A CAS user usually does not *see* the bundle of tracks, although some systems provide the user an option to view randomly selected tracks to obtain a view of the flow of the problem.

FIGURE 8.5. INPUTS TO TARGET MOTION ILLUSTRATION

Scenario		Position at time 0				Course				Speed (kts)			
#	Probability	*A*	*B*	*C*	*D*	060T	075T	090T	105T	8	9	10	11
I	.6	.7	.3	*	*	*	.8	*	.2	.4	*	.6	*
II	.4	*	*	.6	.4	.5	*	.5	*	*	.7	*	.3

Track	Scenario	Position at time 0	Course	Speed (knots)	Initial track weight (probability)
1	I	*A*	075T	8	0.134
2	I	*A*	075T	10	0.202
3	I	*A*	105T	8	0.034
4	I	*A*	105T	10	0.050
5	I	*B*	075T	8	0.058
6	I	*B*	075T	10	0.086
7	I	*B*	105T	8	0.014
8	I	*B*	105T	10	0.022
9	II	*C*	060T	9	0.084
10	II	*C*	060T	11	0.036
11	II	*C*	090T	9	0.084
12	II	*C*	090T	11	0.036
13	II	*D*	060T	9	0.056
14	II	*D*	060T	11	0.024
15	II	*D*	090T	9	0.056
16	II	*D*	090T	11	0.024
					1.000

The user does see on request a probability map pertaining to a given time, e.g., as in Figures 8.7 and 8.8, pertaining to times 0 and 3 hours. The probabilities might be color coded rather than be presented as numbers. To find the probability that the target is in a given cell of a map, the program determines which tracks have the target position in the chosen cell *at the map time* and simply adds the probabilities of those tracks to obtain the cell probability. In Figure 8.7 the prior distribution of the target position is given, taken directly from the scenario weights and initial position distributions of Figure 8.5. In Figure 8.8 the map is derived from moving the target along each track at the speed of that track for 3 hours.

FIGURE 8.6. ILLUSTRATIVE MONTE CARLO TARGET MOTION MODEL

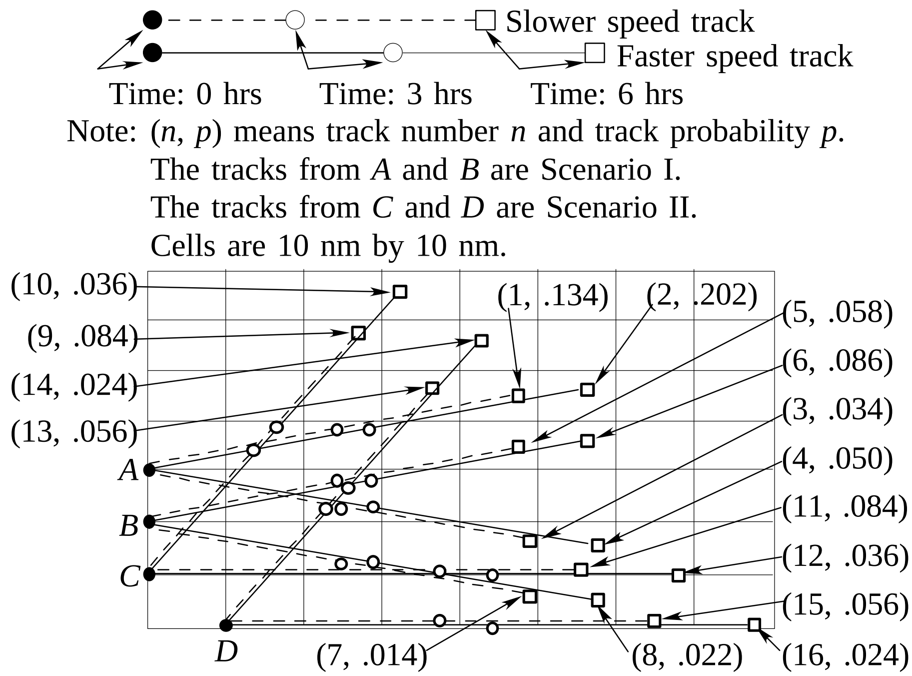


FIGURE 8.7. TARGET POSITION PROBABILITY MAP (TIME = 0)

Convention: A cell boundary point is considered in the cell above or to the right of the boundary. Cells are 10 nm by 10 nm.

	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
A	.42	0	0	0	0	0	0	0
B	.18	0	0	0	0	0	0	0
C	.24	0	0	0	0	0	0	0
	0	.16	0	0	0	0	0	0

D

An important remark is to be made about generating a track bundle by sampling. In the 16-track example it is feasible to use *all* the points in the four two-point distributions, in fact with such small sample spaces there is no reasonable alternative. When a more realistic number of choices of course, speed, changes in these, etc., are available, a bundle of tracks (typically 500 or more) is usually constructed by *sampling* each of the distributions of course, speed, etc. This is where the method really becomes Monte Carlo. When the construction is by sampling, the initial weight (probability) is assigned to be the same for all tracks. The probability structure over the track bundle comes from the relative densities of tracks, i.e., there will be relatively numerous tracks with high-probability courses, etc. As the search progresses without success, the track weights will shift, as described below.

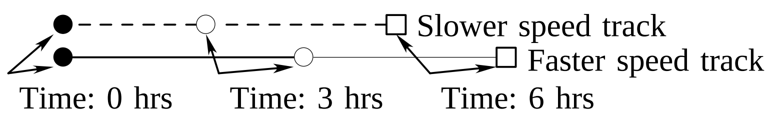
FIGURE 8.8. TARGET POSITION PROBABILITY MAP (TIME = 3 HRS)

Convention: A cell boundary point is considered in the cell above or to the right of the boundary. Cells are 10 nm by 10 nm.

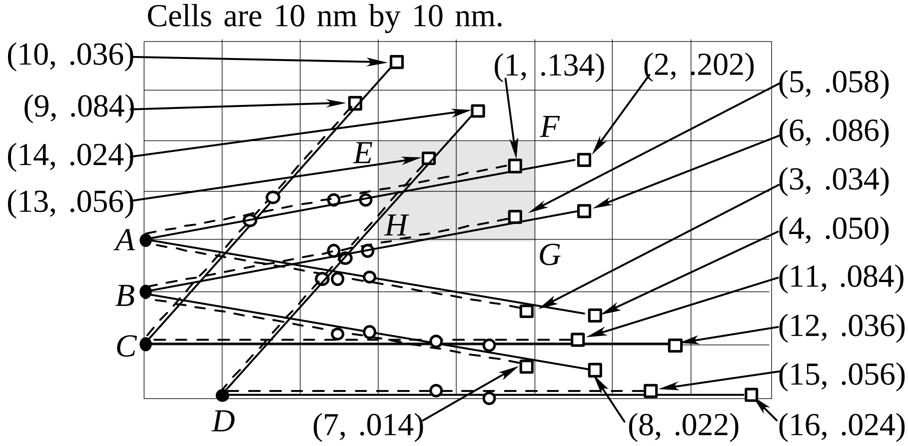
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	.120	.336	0	0	0	0	0
0	0	.308	0	0	0	0	0
0	0	.120	.036	0	0	0	0
0	0	0	.056	.024	0	0	0

A *B* *C* *D*

FIGURE 8.9. APPLICATION OF SEARCH EFFORT



Note: (*n*, *p*) means track number *n* and track probability *p*.
The tracks from *A* and *B* are Scenario I.
The tracks from *C* and *D* are Scenario II.
Cells are 10 nm by 10 nm.



Updates for new information. Now updating for negative information is illustrated. Suppose that from time 3 hours to time 6 hours search effort is applied uniformly over the square shown in

Figure 8.9 as *EFGH*. Suppose that as of time 6 hours no detection has been made, and it is desired to update the probability map to reflect this negative information. First one needs an estimate of the

FIGURE 8.10. CUMULATIVE DETECTION PROBABILITY

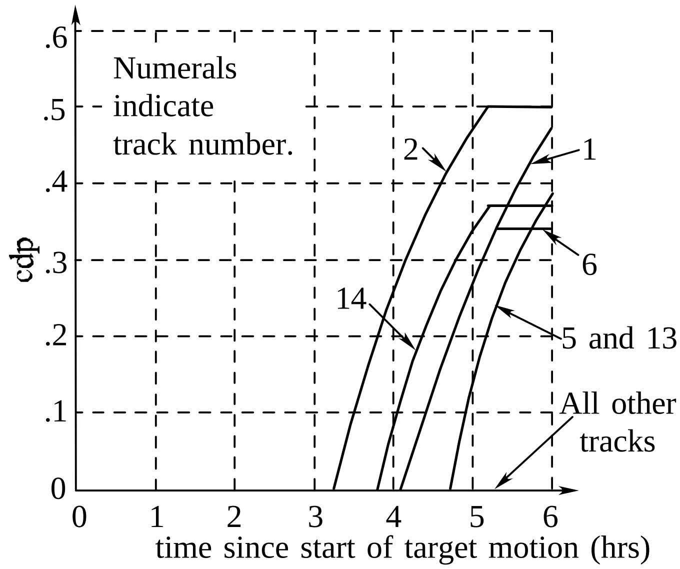


FIGURE 8.11. UPDATE FOR NEGATIVE INFORMATION

Search effort is applied uniformly over rectangle *EFGH* from time 3 hours to time 6 hours. No detection occurs. What are the inferred new (posterior) track weights?

[1] Track # *i*	[2] Pre-search (prior) track weight (normalized)	[3] Search failure probability if track *i* is actual	[4] [2]x[3] = posterior track weight (unnormalized)	[5] Normalized weight [4]/*S*
1	0.134	0.52	0.070	0.093
2	0.202	0.50	0.101	0.134
3	0.034	1.00	0.034	0.045
4	0.050	1.00	0.050	0.066
5	0.058	0.60	0.035	0.046
6	0.086	0.67	0.058	0.077
7	0.014	1.00	0.014	0.019
8	0.022	1.00	0.022	0.029
9	0.084	1.00	0.084	0.112
10	0.036	1.00	0.036	0.048
11	0.084	1.00	0.084	0.112
12	0.036	1.00	0.036	0.048
13	0.056	0.60	0.034	0.045
14	0.024	0.64	0.015	0.020
15	0.056	1.00	0.056	0.074
16	0.024	1.00	0.024	0.032
	1.000		*S* = 0.753	1.000

FIGURE 8.12. TARGET POSITION PROBABILITY MAP (TIME = 6 HOURS AFTER NEGATIVE INFORMATION)

Convention: A cell boundary point is considered in the cell above or to the right of the boundary. Cells are 10 nm by 10 nm.

0	0	0	.048	0	0	0	0
0	0	.112	0	.020	0	0	0
0	0	0	.045	.093	.134	0	0
0	0	0	0	.046	.077	0	0
0	0	0	0	0	0	0	0
0	0	0	0	.045	.178	.048	0
0	0	0	0	.019	.029	.074	.032

A B C D

FIGURE 8.13. TARGET POSITION PROBABILITY MAP IF NO SEARCH WERE MADE (TIME = 6 HOURS)

Convention: A cell boundary point is considered in the cell above or to the right of the boundary. Cells are 10 nm by 10 nm.

0	0	0	.036	0	0	0	0
0	0	.084	0	.024	0	0	0
0	0	0	.056	.134	.202	0	0
0	0	0	0	.058	.086	0	0
0	0	0	0	0	0	0	0
0	0	0	0	.034	.134	.036	0
0	0	0	0	.014	.022	.056	.024

A B C D E F G H

effectiveness of the search effort cell by cell, and one must combine that with the assumptions of target motion track by track. As to search effectiveness, one must find a curve of cumulative detection probability (cdp) *for each track*. This is illustrated in Figure 8.10. In most CAS systems this is done by a (λ, σ) model (Chapter 5). The negative information update is now applied to the track probabilities, as shown in Figure 8.11. This again applies Bayes' theorem in analogy to 801, where the updating is on cell probabilities. Column [2], the prior for the current Bayesian update, is obtained from the track probabilities in Figure 8.5. Column [3], the likelihoods, is obtained by complementing the 6-hour probabilities in Figure 8.10. Column [4] is the product of columns [2] and [3] and is proportional to the posterior track probabilities at time 6 hours. The posterior probabilities, column [5], are obtained by normalizing column 4 and reflect the 3 hours of unsuccessful search as desired. The posterior distribution over the tracks is translated into the posterior distribution over the cells by the method used to produce Figure 8.8. This results in Figure 8.12 which is the probability map for time 6 hours, reflecting the 3 hours of unsuccessful search as well as 6 hours of target motion. Figure 8.13 shows what the distribution of target position would have been at time 6 hours if there had been no search. Note that in the searched rectangle, *EFGH*, the unsuccessful search drove the probabilities lower in Figure 8.12 than in Figure 8.13, but not to 0, except for the cell that was 0 initially. Note also that in the two cells to the east of *EFGH* and in one cell to the north the probabilities also decreased; that is because the tracks with positions in these cells at 6 hours had *passed through EFGH* while it was being (unsuccessfully) searched. To offset these decreases, in all other cells the probability is higher in Figure 8.12 than in Figure 8.13.

This completes the description of updating for negative information in Monte Carlo moving target CAS. As review, the use of Bayes' theorem is analogous to the stationary case, but the updated probabilities pertain to tracks rather than positions. The track probabilities are readily converted to position probabilities at any desired time.

804 Optimal Search for a Moving Target

In this section an algorithm is given to allocate search effort optimally against a moving target. After the algorithm is specified, an example illustrates its mechanism.

The target motion is now assumed to be a Markov chain moving among a finite set C of cells at search times $i = 0, 1, \ldots, n$. (Treatment of other types of motion such as that of 803 is noted at the end of the section.) An optimal plan (allocation of search effort in space and time) maximizes probability of detection by time n. The target occupies one cell at a given time. It is also assumed that search effectiveness is exponential as defined in the stationary search algorithm in 801, final subsection, using β as specified there. One unit of search effort is available at each search time and may be allocated over the cells as the searcher desires. It is easy to generalize what follows to let β and the available effort depend on time. With added complication, β may depend also on cell.

The algorithm is a sequence of iterations indexed $m = 1, 2, \ldots$; iteration m outputs a search plan x^m and the probability Q^m that the search *fails* at every search time and cell during the iteration. Here $x_i^m(c)$ is the amount of search effort that the plan x^m applies to cell c at time i. For each $m > 1$, iteration m is at least as good as its predecessor: $Q^m \le Q^{m-1}$. The sequence of plans $(x^1, x^2, \ldots)$ converges to a limit. The limit plan minimizes the probability that detection fails at every time and cell. For initiation of iteration 1, one may choose an arbitrary plan, x^0; this may be thought of as an output of an artificial iteration 0. For example, one might choose $x_i^0(c) = 0$ for c in C and $i = 0, \ldots, n$. If that choice is made, then iteration 1 will output a myopic plan, i.e., one that does not look ahead in search times.

It is interesting that the plan produced in a given iteration allocates effort optimally, at each search time i, against a *stationary* target, *given* non-detection at all times *before i* using the plan of the *current* iteration and *given* non-detection at all times *after i* using the plan of the *previous* iteration. That is the heart of the algorithm. It reduces the problem of allocating effort in space and time to a time sequence of problems of instantaneous allocation over space against a stationary target (whose position distribution is suitably conditioned as just noted). An algorithm to solve such a stationary-target problem has been given at the end of 801. Recall that algorithm is composed of “application steps.” These are within a search time of the present algorithm, and the search times are within an iteration. One must not confuse these three sequences with one another.

Additional inputs are, for c, d in C and $i = 0, \ldots, n$,

$$g(c) \equiv \Pr\{\text{cell } c \text{ contains the target at search time } 0\},$$

$$\tau(c, d) \equiv \Pr\{\text{the target is in cell } d \text{ at a given search time} \mid \text{it is in cell } c \text{ at the preceding search time}\}.$$

The containment probabilities will shift between search times because of Bayesian updating for unsuccessful search and because of target motion. It is assumed that containment and search effectiveness probabilities are independent of cell, search time, and each other.

Specification of the algorithm. The algorithm initiates by a choice of x^0 as above. For $m > 0$, iteration m proceeds by computing a backward recursion for search times $i = n$ down to $i = 0$:

$$b_i(c) \equiv \Pr\{\text{a target in cell } c \text{ at time } i \text{ will not be detected by search after time } i\}$$

$$= \begin{cases} 1, & \text{if } \quad i = n, \\ \sum_{d \text{ in } C} \tau(c, d) \exp(-\beta x_{i+1}^{m-1}(d)) b_{i+1}(d), & \text{if } \quad i < n, \end{cases} \tag{8-1}$$

for c in C.

$$f_i(c) \equiv \Pr\{\text{the target is in cell } c \text{ at time } i \text{ and was not detected by search before time } i\}$$

$$= \begin{cases} g(c), & \text{if } \quad i = 0, \\ \sum_{d \text{ in } C} f_{i-1}(d) \exp(-\beta x_{i-1}^{m}(d)) \tau(d, c), & \text{if } \quad i > 0. \end{cases} \tag{8-2}$$

The iteration next does a forward recursion for search times $i = 0, \ldots, n$. Fix i, $0 \le i \le n$. Compute

$$s_i(c) \equiv \Pr\{\text{the target is in cell } c \text{ at search time } i \text{ and is not detected by search at any time other than } i\} \tag{8-3}$$

$$= f_i(c) b_i(c),$$

for c in C. One can now compute, with i still fixed,

$$x^m = \text{optimal search plan for a stationary target with defective distribution } s_i \text{ using one unit of search effort, computed as in 801.} \tag{8-4}$$

In (8-2), "search before time i" refers to plans x_0^m at time 0, x_1^m at time 1, ... , x_{i-1}^m for time i - 1. Similar remarks apply to (8-1) on b, referring to plans at *future* times from the *previous* iteration, m - 1, and to (8-3) on s.

Unless $i = n$, one replaces i by i + 1 and proceeds with the forward recursion. If $i = n$, the forward recursion is complete and so is iteration m. As the iterations progress, the plans x^1, x^2, ... converge to a single, and optimal, plan. The failure probability, Q^m, under plan x^m, is given by

$$Q^m = \text{probability of non-detection using plan } x^m = \sum_{c \text{ in } C} s_n(c) \exp(-\beta x_n^m(c)). \tag{8-5}$$

The next iteration ensues. The algorithm stops when Q^{m-1} - Q^m becomes less than a pre-assigned threshold.

Example. In this example, C consists of three cells, indexed $c = 1, 2, 3$. There are two search times, 0 and $1 = n$. The target's prior position, i.e., containment, distribution is $g(1) = .6$, $g(2) = .4$, and $g(3) = 0$. If the target is in cell 1 or 2 and if z amount of search effort is applied to that cell, then detection probability is $1 - \exp(-1.5z)$, thus $\beta = 1.5$. When the target is in cell 3, it is undetectable, so cell 3 is excluded from allocations of effort; only 0 effort is applied to cell 3. Also, between successive search times, if the target is in cell 1, it moves to cell 2, if in cell 2, it moves to cell 3, and if in cell 3, it stays there; thus $\tau(1, 2) = \tau(2, 3) = \tau(3, 3) = 1$, while otherwise $\tau(c, d) = 0$. At each search time, 1 unit of search effort is available.

Although it is quite instructive to carry out the lengthy hand computation of the algorithm for this example, in this particular case a much easier solution is available by calculus – see problem 10. Examples with substantially greater numbers of cells and search times are generally not practical to compute by hand, and generally they do not have convenient alternative solution methods.

The computation of the algorithm will be presented as a sequence of numbered statements, followed by explanation or other comments. Attention is particularly directed to statement (14).

Initiation by artificial iteration 0:

(1) Set $x_i^0(c) = 0$, for $i = 0, 1$, $c = 1, 2, 3$.

Iteration 1:

Time 1:

(2) Set $b_1(1) = b_1(2) = b_1(3) = 1$.

Applies (8-1), noting that $1 = n$.

Time 0 (time has moved backward):

(3) Set $b_0(1) = b_0(2) = b_0(3) = 1$.

Applies (8-1), (1), and (2). Note that for each c, in the summation in (8-1) there is one term (one d) with $\tau(c, d) = 1$ and the other τ factors are 0.

(4) Set $f_0(1) = .6$, $f_0(2) = .4$, $f_0(3) = 0$.

Applies (8-2). This is the prior containment distribution.

(5) Set $s_0(1) = .6$, $s_0(2) = .4$, $s_0(3) = 0$.

Applies (8-3), (4), and (3).

(6) Compute $x_0^1(1) = .635$, $x_0^1(2) = .365$, $x_0^1(3) = 0$.

Applies (5) and 801, last subsection, to find best allocation of 1 unit of effort over cells 1 and 2 with containment probabilities .6 and .4 respectively. Application step 1 applies $(1/1.5)\ln(.6/.4) = .270$ to the higher probability cell, cell 1, since that does not exhaust 1, and 0 effort to cell 2.

For application step 2, necessarily the containment probabilities for cells 1 and 2 are equal, even without computing $S = .4 + .4 = .8$, so that step divides the remaining effort between the two cells: $.635 = .270 + (1 - .270)/2$, and $.365 = 1 - .635$.

Time 1 (time has moved forward):

(7) Compute $f_1(1) = 0$, $f_1(2) = .6e^{-1.5\times.635} = .231$, $f_1(3) = .4e^{-1.5\times.365} = .231$.

Applies (8-2), (4), and (6). For any cell d, $\tau(d, 1) = 0$. For $c = 2$ or 3, a remark on τ similar to (3) applies here also.

(8) Set $s_1(1) = 0\times1 = 0$, $s_1(2) = .231\times1 = .231$, $s_1(3) = .231\times1 = .231$.

Applies (8-3), (7), and (2).

(9) Set $x_1^1(1) = 0$, $x_1^1(2) = 1$, $x_1^1(3) = 0$.

Applies (8-4) and (8). This does not need 801 because $s_1(1) = 0$ and the target is undetectable in cell 3, so all effort must be applied to cell 2.

(10) Compute $Q^1 = 0 + .231e^{-1.5\times 1} + .231\times 1 = .283$.

Applies (8-5), (8), and (9). Thus .283 is the failure probability for myopic search over the two search times.

Iteration 2:

Time 1:

(11) Set $b_1(1) = b_1(2) = b_1(3) = 1$.

Same as (2).

Time 0:

(12) Compute $b_0(1) = e^{-1.5\times 1} = .223$, $b_0(2) = b_0(3) = 1$.

Applies (8-1) and, for $c = 1$, (9), otherwise same as (3).

(13) Set $f_0(1) = .6$, $f_0(2) = .4$, $f_0(3) = 0$.

Same as (4).

(14) Compute $s_0(1) = .6\times.223 = .134$, $s_0(2) = .4\times 1 = .4$, $s_0(3) = 0$.

Applies (8-3), (13), and (12). *Important note*: At this point the conditioned containment probabilities *at time* 0 shift so the majority is in cell 2 rather than cell 1. That is how the computed plan improves over the myopic plan. It results from the time 0 backward recursion in this iteration, i.e., (12).

(15) Compute $x_0^2(1) = .135$, $x_0^2(2) = .865$, $x_0^2(3) = 0$.

Similar to (6), applies (8-4), 801, and (14).

Time 1:

(16) Compute $f_1(1) = 0$, $f_1(2) = .6e^{-1.5\times.135} = .490$, $f_2(3) = .4e^{-1.5\text{x}.865} = .109$.

Similar to (7), applies (8-2), (13), and (15).

(17) Set $s_1(1) = 0$, $s_1(2) = .490$, $s_1(3) = .109$.

Applies (8-3), (16), and (11).

(18) Set $x_1^2(1) = 0$, $x_1^2(2) = 1$, $x_1^2(3) = 0$.

Same as (9).

(19) Compute $Q^2 = 0 + .490\text{e}^{-1.5} + .109 = .219$.

Similar to (10).

It is easily confirmed that if iteration 3 is undertaken, it repeats iteration 2. Therefore the algorithm stops at this point. For this example, it provides the optimal allocation of unit effort over the two search times: At search time 0, apply .135 amount of effort to cell 1, .865 to cell 2, and none to cell 3, and at time 1 apply the entire unit of effort to cell 2. The myopic plan is the same for time 1, but at time 0 applies .635 to cell 1 and .365 to cell 2. The failure probability is $Q^1 = .283$ for the myopic plan and is $Q^2 = .219$ for the optimal plan, a 23 percent improvement. In many, perhaps most, realistic moving target search allocation problems a myopic plan is much closer to optimality than in this example. As one can see from this relatively simple example, the algorithm is rather lengthy to compute by hand, but it is not difficult to program for a computer.

Suppose that target motion is given as a track bundle, call it B, as illustrated in 803, instead of assuming that it is a Markov chain. The above algorithm might be adapted to such motion. To illustrate, suppose the problem is to plan placement of a sonobuoy field (a timed collection of patterns) during each of a sequence of VP ASW patrol sorties. Regard each sortie as a time step. Let the assignment of effort on sortie i be determined by choice of a timed point, call it x_i, at which to anchor the buoy field for that sortie. For any track T in B, let $g(T)$ be the pre-search probability that T is the correct track; let $u_i(x_i, T)$ be the probability that if T is the correct track, and x_i is chosen to anchor the field on sortie i, then detection will not occur on that sortie. Because the field has a lifetime, to evaluate u_i, a cdp model, e.g., Figure 8.10, is needed. As before, iterations will be computed, indexed by m.

Now define b_i, f_i, and s_i as before, but using tracks in place of cells, somewhat in analogy to the cell definitions: For each track T in B, and for $i = n$ down to 0, let

$$b_i(T) = \Pr\{\text{if } T \text{ is the correct track then no detection occurs after sortie } i\}$$

$$= \begin{cases} 1 & \text{if } i = n, \\ b_{i+1}(T)u_{i+1}(x_{i+1}^{m-1}, T), & \text{otherwise.} \end{cases}$$

Next, for $i = 0, \ldots, n$ and T in B, let (as before iterations are indexed by m)

$$f_i(T) = \Pr\{T \text{ is the correct track and no detection occurs before sortie } i\}$$

$$= \begin{cases} g(T) & \text{if } i = 0, \\ f_{i-1}(T)u_{i-1}(x_{i-1}^{m}, T) & \text{otherwise,} \end{cases}$$

$$s_i(T) = \Pr\{T \text{ is correct and no detection on other than sortie } i\}$$

$$= b_i(T)f_i(T),$$

$$x^{m} = \text{optimal anchor point for buoy field on sortie } i \text{ given that the defective track distribution is } s_i.$$

The plans $x^1, x^2, \ldots$ converge to a limit plan, which is approximately optimal. This leaves unaswered the question of how to compute an optimal anchor point for single sortie and a given distribution of tracks, but it does reduce the multiple-sortie problem to a sequence of single-sortie problems. The single-sortie problems may be generalized to choice of optimal *configuration* of the buoy field, in addition to optimization of its placement.

805 Other Literature and History

Figure 8.2, was taken from H. R. Richardson and L. D. Stone [1]. The algorithm described at the end of 801 for optimal allocation of search against a stationary target is given more formally in 2.2.8 of Stone [2, 3], where it is credited to A. Charnes and W. W. Cooper [4]. Monte Carlo CAS, illustrated in 803, was originated by Richardson; the illustration is taken from Wagner [5].

The exposition of the optimal moving target search algorithm given in 804 follows S. S. Brown [6] with different notation. The three-cell example in 804 illustrating that algorithm and the two-cell example in Problem 11a were inspired by A. R. Washburn's example in 6.2 of [7], which illustrates dramatically that myopic search need not be optimal. In the example of [7], at a given time instant all of the effort must be placed in a single cell.

The first operational CAS system was the U.S. Coast Guard search and rescue program, Computer-Assisted Search Planning, developed 1970-72 under Richardson [8]. LCDR J. H. Discenza, USCG, was instrumental in implementation of CASP (see [9]). The multi-scenario method of constructing a prior was originated by J. P. Craven in the 1966 H-bomb search off Palomares, Spain. Perhaps the first application of Bayes' theorem to update for negative information in actual search planning was by Richardson on scene in that operation (without electronic computation). A multi-scenario prior was again applied in the *Scorpion* search in 1968, as noted in 801; this time electronic computation was used (remotely). Bayesian updating of the prior was done by hand on the search scene.

Richardson and his colleagues extended these methods to develop and to apply successfully various ASW CAS systems in the 1970s. The first desktop-calculator and seagoing CAS system was developed by T. L. Corwin at COMSUBPAC in 1975. It used analytic rather than Monte Carlo methods.

The progress of the 1970s culminated in development of VPCAS under COMPATWINGSPAC and COMPATWINGSLANT, led by S. J. Benkoski and R. P. Buemi. VPCAS was a Monte Carlo program to assist mission planning in VP ASW and was introduced to ASW Operations Centers (ASWOCs) in late 1983. Successor extensions, developed primarily by W. R. Monach, were the search and tracking systems PACSEARCH, under COMOCEANSYSPAC, and the OCAS module [10] in the ONR system OPTAMAS. These programs included algorithms for optimal multi-sortie sonobuoy search along the lines of that given at the end of 804. The 1987 prototype SALT was developed primarily by Stone, D. A. Trader, and Corwin and evolved into Nodestar [11]; these systems used Markov-chain target-motion modeling and recursive Bayesian filtering methods [12], which make combined inference from positive and negative information. OCAS and Nodestar are much more sophisticated than the CAS systems of the early 1970s, and their hardware hosts are much more powerful.

The algorithm for optimal search for moving targets presented in 804 was given in 1977 by Brown [13]; see also Brown [14] and Stone, et al., [15]. Assuming that the amount of effort allocable to a given cell is arbitrary and effectiveness is exponential, both as assumed here, he showed that for a plan to be optimal it is necessary and sufficient that all instantaneous allocations, conditioned on non-detection in the future and past, be optimal. He proved that under arbitrary discrete target motion, the algorithm converges to a limit plan and that the limit plan satisfies this instantaneity condition.

Although the algorithm in 804, sometimes called “Brown's algorithm” and sometimes the “FAB algorithm” (for forward and backward), is extremely interesting and sometimes useful, in most practical applications a myopic plan is quite close to optimality. In [16], Washburn showed that even if at a given instant all effort must be allotted to a single cell, then the above instantaneity condition is still necessary (true for any restriction on the amounts allocable to a single cell), and showed by example that sufficiency may fail. In [17] Washburn gave a bound which enables one to tell when a solution comes within a given ϵ of optimal detection probability. In [18] he extended the algorithms of [15] and [16] to payoffs other than probability of detection in specified time. For further results and history on optimal moving target search see Appendix C of [3].

Richardson and Corwin in reference [19] is an article on the principal methods used in Monte Carlo CAS. An excellent tutorial on CAS, with very modest mathematical prerequisites, is given in [20]. It contains an eight-track example similar to the above sixteen-track example, but shows only what the user would see and not the tracks themselves. References Koopman [21, 22], Stone [2, 3], and Washburn [7] are general texts on methods in search analysis. Reference [2] was awarded the Lanchester Prize of the Operations Research Society of America as the best 1975 publication in English on operations research. Stone it reference [23] describes the search planning processes. Richardson et al., [24] is a manual for analysis of deep ocean search; among other things it gives guidance for choosing cell size. More detailed history and some elaboration on VPCAS, PACSEARCH, and SALT are given in Wagner [5]. Extensive bibliographies of search literature are given in Benkoski [25] and Stone [2, 3].

Koopman's pioneering role in search theory and its applications is noted in Chapter 6. Craven was Technical Director of the 1966 H-bomb search and the 1968 *Scorpion* search, while he was Chief Scientist at Special Projects, the developers of Polaris missiles and submarines. Richardson, Stone, Corwin, Benkoski, Buemi, and Monach, were with Daniel H. Wagner, Associates, during their cited contributions, except for the work on SALT through Nodestar by Stone, Trader, and Corwin, which was at Metron, Inc. Discenza's work on CASP was at the USCG Rescue Coordination Center, Governor's Island, N.Y. Washburn's various contributions were as an operations research professor at the Naval Postgraduate School. Andrews commanded the *Thresher* search in 1963-64 as CAPT, USN, and consulted on scene to the H-bomb and *Scorpion* searches. Stone [26] details the search planning done for the search for the SS Central America, a side-wheel steamer that sunk in 1857 with over 3 tons of gold. The use of the multi-scenario approach for obtaining the initial probability map is laid out in detail.

[1] Richardson, Henry R., and Lawrence D. Stone. “Operations Analysis During the Underwater Search for Scorpion.” *Naval Logistics Research Quarterly* 18 (June, 1971):141-157.

[2] Stone, Lawrence D. *Theory of Optimal Search*. New York: Academic Press, 1975.

[3] Stone, Lawrence D. *Theory of Optimal Search*. 2nd edition. Baltimore: Operations Research Society of America, 1989.

[4] Charnes, A., and W. W. Cooper. “The Theory of Search: Optimum Distribution of Search Effort,” *Management Science* 5 (1958): 44-50.

[5] Wagner, Daniel H. *Naval Tactical Decision Aids*. Military Operations Research Lecture Notes, NPSOR-1. Monterey CA: Naval Postgraduate School, 1989.

[6] Brown, Scott S. “Optimal Search: Markov Chain Motion, Exponential Effectiveness Function.” *Search, Detection, and Tracking*. Vol. II, Area 2, *Military OR Analyst’s Handbook*. Alexandria, VA: Military Operations Research Society (to be published).

[7] Washburn, Alan R. *Search and Detection*. Baltimore, MD: Operations Research Society of America, 1981.

[8] Richardson, Henry R. *Final Report on Computer-Assisted Search Planning System* (*CASP*). Paoli, PA: Daniel H. Wagner, Associates Memorandum to Commandant, U. S. Coast Guard Headquarters, July 17, 1975.

[9] Richardson, Henry R., and Joseph H. Discenza. “The United States Coast Guard Computer-Assisted Search Planning System (CASP).” *Naval Research Logistics Quarterly* 27

(December, 1980): 659-680.

[10] Director ASW Environmental Acoustics Support Program. *OPTAMAS Version 2.0 System Description*. AEAS Report 900-001. Office of Naval Research, March, 1990.

[11] Stone, Lawrence D.; Thomas L. Corwin; and James B. Hoffman. *Technical Documentation of Nodestar*. Reston, VA: Metron, Inc. Report to the Naval Research Laboratory, December 11, 1995 (DTIC No. AD-A302 458/XAG).

[12] Corwin, Thomas L., and Lawrence D. Stone. "Bayesian Filtering." *Search, Detection, and Tracking*. Vol. II, Area 2, *Military OR Analyst's Handbook*. Alexandria, VA: Military Operations Research Society (to be published).

[13] Brown, Scott S. *Optimal and Near Optimal Search for a Target with Multiple Scenario Markovian, Constrained Markovian, or Geometric Memory Motion in Discrete Time and Space*. Paoli, Pa: Daniel H. Wagner, Associates Memorandum Report to the Office of Naval Research, June 14, 1977.

[14] Brown, Scott. S. "Optimal Search for a Moving Target in Discrete Time and Space." *Operations Research* 28 (November-December, 1980):1275-1289.

[15] Stone, Lawrence. D.; Scott S. Brown; Robert. P. Buemi; and Carol R. Hopkins. *Numerical Optimization of Search for a Moving Target*. Daniel H. Wagner, Associates Report to the Office of Naval Research, 1978 (DTIC No. AD-A058 470/XAG).

[16] Washburn, Alan R. "On Search for a Moving Target." *Naval Research Logistics Quarterly* 27 (June, 1980): 315-322.

[17] Washburn, Alan R. "An Upper Bound Useful in Optimizing Search for a Moving Target." *Operations Research* 29 (November-December, 1981): 1227-1230.

[18] Washburn, Alan R. "Search for a Moving Target: the Fab Algorithm." *Operations Research* 31 (July-August, 1983): 739-751.

[19] Richardson, Henry R., and Thomas L. Corwin. "An Overview of Computer-Assisted Search." Haley, K. Brian, and Stone, Lawrence D., eds., *Search Theory and Applications*. New York: Plenum, 1980.

[20] Benkoski, Stanley J. *Introduction to ASW and Computer-Assisted Search*. Commander Patrol Wings, Pacific Fleet Tactical Study sz033-z-83, 26 September, 1983.

[21] Koopman, Bernard O. *Search and Screening*. Operations Evaluation Group Report 56. Washington DC: Office of the Chief of Naval Operations, 1946.

[22] Koopman, Bernard O. *Search and Screening: General Principles with Historical Applications*. New York: Pergamon Press, 1980.

[23] Stone, Lawrence D. "The Process of Search Planning: Current Approaches and Continuing Problems." *Operations Research* 31 (March-April, 1983): 207-233.

[24] Richardson, Henry R.; Lawrence D. Stone; and Frank A. Andrews. *Manual for Operations Analysis of Deep Ocean Search*. Supervisor of Salvage, Naval Ships Systems Command Publication 0994-010-7010, 1971.

[25] Benkoski, Stanley J.; Michael G. Monticino; and James R. Weisinger. "A Survey of Search Theory Literature." *Naval Research Logistics*. 38 (August 1991): 469-494.

[26] Stone, Lawrence D. "Search for the SS Central America: Mathematical Treasure Hunting." *Interfaces* 22 (January-February 1992): 32-54.

Problems

1. A stationary object has been lost in cell 1, 2, or 3. Scenarios I and II have been postulated for the loss. A priori under I, it is twice as probable that the object is in cell 2 as in cell 1 and three times as probable that it is in cell 3 as in cell 1. Under II, all three cells are equally likely to contain the target. Scenario II is deemed twice as likely as I.

a. Under the composite of I and II, which cell has the highest probability of containing the target?
b. What is that probability?
c. Why is no calculation needed to answer a.?

2. A stationary object is lost at a point on a line with a coordinate scale and origin. There are two equally likely scenarios for the loss, I and II. The distribution of the position is normal for both. For I and II respectively, the means are 0 and 2 and the variances are 4 and 9. Under the composite of these scenarios, what is the probability that the object is at a negative coordinate?

3. For the optimization example in 801, answer the following:

a. After the 1.02 hours of search effort have been applied to (2, 2), how much additional search time should be applied to (2, 2), (1, 2), and (2, 1) before effort should begin in (3, 3), (2, 3), and (3, 2)?
b. What is the distribution of position at that point?
c. What computation saving in CAS programs is suggested by this exercise?

4. Why are the track probabilities the same in Figures 8.6 and 8.9, while the cell probabilities in Figures 8.7 and 8.8, at the corresponding times, are different?

5. In the prior distribution of tracks in Figure 8.5, find the distributions of

a. target course and
b. target speed.

6. As of the update in Figure 8.11, compute the posterior distributions of

a. target speed,
b. target course, and
c. target scenarios.
d. How has inference from unsuccessful search changed one's view of the scenarios compared to the pre-search situation?

7. Using Figures 8.10 and 8.11, given unsuccessful search through time 4 hours, what is the probability that the actual track is number 1?

8. A target moves among the following nine boxes:

1	2	3
4	5	6
7	8	9

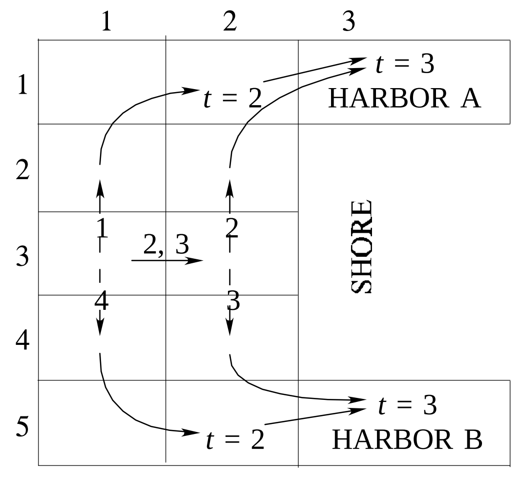
There are four tracks the target can take, given as follows, with the stated probabilities:

Track Number	Box number at time 0	Box number at time 1	Box number at time 2	Prior probability that the track occurs
1	1	2	3	.1
2	1	5	3	.3
3	7	5	9	.2
4	7	8	9	.4

At time 1, box 5 is searched with detection probability .4, given that the target is in box 5 at time 1. The search does not succeed. What is the probability that the target is in box 9 at time 2?

9. A momentary contact on an enemy submarine occurred in cell (3, 1), shown below. Note that cell (i, j) is row i, column j. It is postulated that the submarine is en route to one of two harbors to sow a minefield. A bottom chart of the area is examined in detail resulting in the postulation of two possible tracks to each of the two harbors, totaling four tracks. Their a priori probabilities are also shown in the figure below. The postulated tracks are converted to positions for each of three times t following the initial contact.

Track:	T_1	T_2	T_3	T_4
A priori probability	.4	.2	.1	.3



a. It is now time 1 and searches are conducted in cells (2, 1) and (4, 1) in such a manner that the probability of detection is .7, given the target was in the cell searched. There were no detections. Update the *track* probabilities.

b. Construct a probability map of target location for time 2 using the track probabilities from a.
c. The weather turns bad and further search is impossible. What is the probability that Harbor *B* is the submarine's destination?

10. In the example of 804, observe that the choice of the amount of effort applied to cell 1 at time 0, $x_0(1)$, determines the amount applied to cell 2 at time 0, since no effort is applied to cell 3. Also, at time 1, all available effort is applied to cell 2, since there is 0 probability that the target is in cell 1. Thus $x_0(1)$ determines the entire search plan. Use this fact and calculus to show that the optimal plan is in fact as found by the algorithm in 804.

11. Suppose at time 0, a target is in cell 1 with probability .3 and in cell 2 with probability .7. After 1 unit of search effort is applied at time 0, divided equally between the two cells, the target moves to cell 1 if it was in cell 2 and to cell 2 if in cell 1. Then at time 1, another unit of effort is applied. When effort z is applied to a cell containing the target, detection probability is 1 - exp(-2z), except that at time 1, if the target is in cell 2 it is undetectable.

a. For this problem apply the algorithm of 804 to find the myopic search plan and the probability that this plan finds the target in one of the two search times, 0 and 1.
b. Continue the algorithm to find the optimal plan and its success probability.
c. Use a method similar to that of problem 10 to verify independently (and more easily) that the solution found in b is correct.

Answers to Problems

1. a. Cell 3. b. 7/18. c. Because cell 3 is not bettered under either I or II.

2. .38.

3. a. 3.43 hrs. b. Distribution:

	1	2	3
1	.135	.144	.000
2	.144	.144	.144
3	.000	.144	.144

c. Postpone calculation of normalizing factors in Bayesian analysis until needed. Much useful information can be gained without them.

4. Motion has occurred to change the cell probabilities, but no search has occurred to change the

track probabilities.

5. Distributions:

a.		b.	
Course	Probability	Speed	Probability
060T	.200	8 kts	.240
075	.480	9	.280
090	.200	10	.360
105	.120	11	.120

6. Distributions:

a.		b.		c.	
Speed	Probability	Course	Probability	Scenario	Probability
8	.203	060T	.225	I	.509
9	.343	075	.350	II	.491
10	.306	090	.266		
11	.148	105	.159		
	1.000		1.000		1.000

7. .141.

8. .65.

9. a. The track probability updates are .235, .392, .196, and .176 for tracks 1, 2, 3, and 4, respectively.
b. The probabilities of being in cells (1, 2), (2, 2), (4, 2), and (5, 2) are respectively .235, .392, .196, and .176; all other cells have probability 0.
c. .373.

10. Additional hint: Apply the methods of 801 for optimal search for a stationary target at each time.

11. a. At time 0, put .788 amount of effort in cell 2 and .212 in cell 1. At time 1, put the entire 1 unit of effort in cell 1. Success probability is .876.
b. At time 0, the myopic plan puts .288 amount of effort in cell 1 and .712 in cell 2; at time 1, all effort is in cell 1. Success probability is .808, compared to .876 for the optimal plan.

Chapter 8 INDEX